I CLAIM:

1. A method for testing changes in a software program using a plurality of test cases, wherein the software program comprises a first plurality of execution paths, the method comprising:

identifying one or more changed paths in the first plurality of execution paths;

from the plurality of test cases, identifying one or more test cases that are capable of executing the one or more changed paths; and

executing the one or more of the identified test cases to test the changed path.

2. The method of claim 1, wherein the software program comprises one or more modules, and identifying one or more test cases comprises identifying a changed module and determining whether the changed module causes changes in the execution paths.

3. The method of claim 1, wherein identifying one or more test cases comprises identifying a second plurality of execution paths in the software program and determining the difference between the first and second pluralities of execution paths.

4. The method of claim 3, wherein the difference comprises at least one

of a new path and a changed path.

5. The method of claim 1, wherein identifying one or more test cases comprises evaluating names of one or more methods of a test case from the plurality of test cases thereby determining whether the methods of the test case involve the one or more changed paths.

6. The method of claim 5, wherein identifying one or more test cases further comprises evaluating parameters of one or more methods of a test case from the plurality of test cases thereby determining whether the methods of the test case involve the one or more changed paths.

7. The method of claim 1, wherein identifying one or more test cases comprises determining whether a test case intersect one or more changed paths.

8. The method of claim 7, wherein determining whether a test case intersect one or more changed execution paths comprises identifying a module of the software program included in both the test case and a changed execution path.

9. The method of claim 8, wherein each module is represented by a node number, and each execution path and test case is represented by a string of node numbers, wherein identifying a module comprises identifying a node number included in both a changed execution path and a test case.

10. A computer program product for testing a software program using a plurality of test cases, the computer program product comprising a computer usable medium having a computer readable program code embodied thereon, the computer readable program code controlling the computer to perform the operations of:

identifying one or more changed paths in a first plurality of execution paths of the software program;

identifying one or more test cases that are capable of executing the one or more changed paths; and

executing the identified one or more test cases to test the changed code of the software program.

11. The computer program product of claim 10, wherein the software program comprises one or more modules, wherein identifying one or more paths comprises identifying the changed module and determining whether the changed module causes changes in the execution paths.

12. The computer program product of claim 10, wherein identifying one or more paths comprises identifying a second plurality of execution paths in the software program upon changing of the code and determining the difference between the first and second pluralities of execution paths.

13. The computer program product of claim 12, wherein the difference

comprises at least one of a new path and a changed path.

14. The computer program product of claim 10, wherein identifying one or more test cases comprises evaluating the names of one or more methods of a test case from the plurality of test cases thereby determining whether the methods of the test case involves the one or more changed paths.

15. The computer program product of claim 14, wherein identifying one or more test cases further comprises evaluating the parameters of one or more methods of a test case from the plurality of test cases thereby determining whether the methods of the test case involve the one or more changed paths.

16. The computer program of claim 10, wherein identifying one or more test cases comprises determining whether a test case intersects one or more changed paths.

17. The computer program of claim 16, wherein determining whether a test case intersect one or more changed execution paths comprises identifying a module of the software program included in both the test case and a changed execution path.

18. The computer program of claim 17, wherein each module is represented by a node number, and each execution path and test case is represented by a string of node numbers, wherein identifying a module comprises identifying a node number included in both a changed execution path and a test

case.

19. A system for testing changes in a software program using a plurality of test cases, wherein the software program comprises a first plurality of execution paths, the system comprising:

means for identifying one or more changed paths in the first plurality of execution paths;

means for identifying one or more test cases from the plurality of test cases that are capable of executing the one or more changed paths,

wherein the one or more identified test cases are executed to test the changed code of the software program.

20. The system of claim 19, wherein the software program comprises one or more modules, wherein upon changing of the code at least one module is changed, and wherein identifying one or more test cases comprises identifying the changed module and determining whether the changed module causes changes in the execution paths.

21. The system of claim 19, wherein identifying one or more test cases comprises identifying a second plurality of execution paths in the software program upon changing of the code and determining the difference between the first and second pluralities of execution paths.

22. The system of claim 21, wherein the difference comprises at least one of a new path and a changed path.

23. The system of claim 19, wherein identifying one or more test cases comprises evaluating names of one or more methods of a test case from the plurality of test cases thereby determining whether the methods of the test case involve the one or more changed paths.

24. The system of claim 23, wherein identifying one or more test cases further comprises evaluating the parameters of one or more methods of a test case from the plurality of test cases thereby determining whether the methods of the test case involve the one or more changed paths.

25. The system of claim 19, wherein identifying one or more test cases comprises determining whether a test case intersects one or more changed paths.

26. The system of claim 25, wherein determining whether a test case intersect one or more changed execution paths comprises identifying a module of the software program included in both the test case and a changed execution path.

27. The system of claim 26, wherein each module is represented by a node number, and each execution path and test case is represented by a string of node numbers, wherein identifying a module comprises identifying a node number included in both a changed execution path and a test case.